



# basic education

---

Department:  
Basic Education  
**REPUBLIC OF SOUTH AFRICA**

**NASIONALE  
SENIOR SERTIFIKAAT**

**GRAAD 12**

**INLIGTINGSTEGNOLOGIE V1**

**NOVEMBER 2016**

**PUNTE: 150**

**TYD: 3 uur**

**Hierdie vraestel bestaan uit 16 bladsye.**

**INSTRUKSIES EN INLIGTING**

1. Hierdie vraestel is in DRIE afdelings ingedeel. Kandidate moet AL DRIE afdelings beantwoord.
2. Die duur van hierdie eksamen is drie uur. As gevolg van die aard van hierdie eksamen is dit belangrik om daarop te let dat jy nie toegelaat sal word om die eksamenlokaal voor die einde van die eksamensessie te verlaat nie.
3. Hierdie vraestel is opgestel met programmeringsterme wat nie programmeringstaal-spesifiek is nie (Delphi/Java (deur die Netbeans IDE te gebruik)).
4. Maak seker dat jy die vrae beantwoord volgens die spesifikasies wat in elke vraag gegee word. Punte sal volgens die voorgeskrewe vereistes toegeken word.
5. Beantwoord slegs wat in elke vraag gevra word. Byvoorbeeld, indien die vraag nie datavalidering vra nie, sal geen punte vir datavalidering toegeken word nie.
6. Jou programme moet op só 'n manier gekodeer word dat dit met enige data sal werk en nie net met die voorbeelddata wat voorsien is of enige data-uittreksels wat in die vraestel verskyn nie.
7. Roetines, soos soek, sorteer en seleksie, moet vanuit eerste beginsels ontwikkel word. Jy mag NIE die ingeboude funksies van 'n programmeringstaal vir enige van hierdie roetines gebruik NIE.
8. Alle datastrukture moet deur jou, die programmeerder, verklaar word, tensy die datastrukture voorsien word.
9. Jy moet jou werk gereeld stoor op die disket/CD/DVD/geheuestokkie wat aan jou gegee is, of op die skyfspasie wat vir hierdie eksamensessie aan jou toegeken is.
10. Maak seker dat jou eksamennummer as kommentaar verskyn in elke program wat jy kodeer, asook op elke gebeurtenis wat aangedui word.
11. Indien dit vereis word, druk die programmeringskode van al die programme/klasse wat jy voltooi het. Jy sal ná die eksamensessie 'n halfuur tyd vir drukwerk gegee word.
12. Aan die einde van hierdie eksamensessie moet jy 'n disket/CD/DVD/geheuestokkie inlewer met al jou werk daarop gestoor OF jy moet seker maak dat al jou werk op die skyfspasie gestoor is wat vir hierdie eksamensessie aan jou toegeken is. Maak seker dat al die lêers gelees kan word.

13. Die lêers wat jy nodig om hierdie vraestel te voltooi, is aan jou gegee op die disket/CD/DVD/geheuestokkie of op die skyfspasie wat aan jou toegeken is. Die lêers word in die vorm van wagwoordbeskermd, uitvoerbare lêers verskaf.

**LET WEL:**

- Delphi-kandidate moet die lêer **DelphiDataAFRNov2016.exe** gebruik.
- Java-kandidate moet die lêer **JavaDataAFRNov2016.exe** gebruik.

Doen die volgende:

- Dubbelklik op die wagwoordbeskermd, uitvoerbare lêer.
- Klik op die 'Extract'-knoppie.
- Sleutel die volgende wagwoord in: **AquaS@&\$16**

Nadat dit onttrek ('extracted') is, sal die volgende lys met lêers in die lêergids **DelphiDataAFRNov2016/JavaDataAFRNov2016** beskikbaar wees:

**DELPHI-LÊERS****Vraag1:**

Vraag1\_P.dpr  
Vraag1\_P.res  
Vraag1\_U.dfm  
Vraag1\_U.pas

**Vraag2:**

DataV2.txt  
Uitstappie\_U.pas  
Vraag2\_P.dpr  
Vraag2\_P.res  
Vraag2\_U.dfm  
Vraag2\_U.pas

**Vraag3:**

Vraag3\_P.dpr  
Vraag3\_P.res  
Vraag3\_U.dfm  
Vraag3\_U.pas

**JAVA (NETBEANS)-LÊERS****Vraag1:**

Vraag1.form  
Vraag1.java

**Vraag2:**

DataV2.txt  
Uitstappie.java  
Vraag2.form  
Vraag2.java

**Vraag3:**

Vraag3.form  
Vraag3.java

**SCENARIO**

Aqua Wonderland is 'n park met 'n watertema wat ontspanning, toegang tot restaurante, inkopies, waterglybane en nog baie ander aktiwiteite bied. Die administrateurs by Aqua Wonderland werk tans aan verskillende tegnieke en strategieë om die park se gewildheid te bevorder.

**AFDELING A****VRAAG 1: ALGEMENE PROGRAMMERINGSVAARDIGHEDE**

Die akwarium by Aqua Wonderland is besig om 'n nuwe, reghoekige haaitenk op te rig. Die administrateurs het 'n sagtewareprogram nodig om die inkomste uit besoekers en die daaglikse uitgawes vir die instandhouding van die tenk te bestuur.

Doen die volgende:

- Kompileer en voer die program uit wat in die **Vraag1**-lêergids voorkom. Die gebruikerskoppelvlak vertoon vyf verskillende afdelings wat Vraag 1\_1 tot Vraag 1\_5 genoem word. Die program het tans geen funksionaliteit nie.

Voorbeeld van grafiese gebruikerskoppelvlak GGK ('GUI'):

## Aqua Wonderland se Haaitenk

**Vraag 1\_1**

Lengte  meter

Breedte  meter

Hoogte  meter

kubieke meter

**Vraag 1\_2**

**Vraag 1\_3**

Leeftyd in maande

**Vraag 1\_4**

Aanvanklike koste

Inkomste gedurende eerste jaar

**Vraag 1\_5**

Kies 'n kursus

Swem

Snorkel

- Voltooi die kode vir elke afdeling van VRAAG 1, soos in VRAAG 1.1 tot VRAAG 1.5 op die volgende bladsy beskryf word.

1.1 **Knoppie [Vraag 1\_1]**

Skryf kode om die lengte, breedte en hoogte van die tenk uit die teksblokkies wat voorsien is, te verkry. Bereken en vertoon die volume van die tenk deur die volgende formule te gebruik:

$$\text{Volume} = \text{lengte} \times \text{breedte} \times \text{hoogte}$$

Voorbeeld van afvoer as die lengte 16.5 meter, die breedte 14.2 meter en die hoogte 12.5 meter is:

Lengte	<input type="text" value="16.5"/>	meter
Breedte	<input type="text" value="14.2"/>	meter
Hoogte	<input type="text" value="12.5"/>	meter
Vraag 1_1	<input type="text" value="2928.75"/>	kubieke meter

(5)

1.2 **Knoppie [Vraag 1\_2]**

Die akwarium betaal vir die water wat gebruik sal word om die tenk mee vol te maak. Gebruik die volume (in kubieke meter) wat in VRAAG 1.1 bereken is om die koste van die water te bereken wat nodig sal wees om die tenk mee vol te maak.

GETAL LITER	KOSTE
Die eerste 500 liter	25 sent per liter
Die volgende 300 liter	35 sent per liter
Die oorblywende liter	45 sent per liter

Voorbeeld van afvoer as die volume 2 928.75 kubieke meter is:

Vraag 1_2	<input type="text" value="R 1 187.94"/>
-----------	---

(9)

1.3 **Knoppie [Vraag 1\_3]**

'n Haai kan slegs vir 'n spesifieke tydperk in 'n tenk bly voordat dit vrygelaat moet word. Die gebruiker moet die leeftyd van 'n haai in 'n tenk in maande insleutel.

Skryf kode om die getal maande wat ingesleutel is, na jare en maande om te skakel en vertoon die jare en maande soos in die voorbeeld hieronder getoon.

Voorbeeld van afvoer as die getal maande wat ingesleutel is, 87 is:

Leef tyd in maande	<input type="text" value="87"/>
Vraag 1_3	<input type="text" value="7 jaar en 3 maande"/>

(6)

#### 1.4 Knoppie [Vraag 1\_4]

Die bestuur wil weet hoeveel jaar dit sal neem om die aanvanklike uitgawe vir die oprig van die haaitenk af te betaal. Die jaarlikse inkomste wat hulle verdien, sal gebruik word om die uitgawe af te betaal. Die aanvanklike bedrag wat bestee is om die haaitenk op te rig en die verwagte inkomste in die eerste jaar, moet ingesleutel word. Na die eerste bedryfsjaar word daar verwag dat die inkomste met 10% per jaar sal verhoog.

Gebruik die waardes wat ingesleutel is om die jaar se nommer, die jaarlikse inkomste en die jaarlikse balans van die oprigtingskoste te bereken en te vertoon. As die aanvanklike koste afbetaal is, vertoon die woord 'Afbetaal' in die 'Balans'-kolom. Alle geldwaardes moet na 'n geldeenheid met twee desimale plekke geformateer wees.

Voorbeeld van afvoer as die koste om die tenk op te rig, R500 000 was en die verwagte inkomste gedurende die eerste jaar R75 000 is:

Aanvanklike koste	500000	
Inkomste gedurende eerste jaar	75000	
Vraag 1_4		
Jaar	Inkomste	Balans
1	R 75 000.00	R 425 000.00
2	R 82 500.00	R 342 500.00
3	R 90 750.00	R 251 750.00
4	R 99 825.00	R 151 925.00
5	R 109 807.50	R 42 117.50
6	R 120 788.25	Afbetaal

(14)

#### 1.5 Knoppie [Vraag 1\_5\_1]

Daar sal 'n openingsdag wees om die voltooiing van die haaitenk te vier. Besoekers wat die openingsdag bywoon, sal die geleentheid kry om aan verskeie pret-aktiwiteite deel te neem. Een van die aktiwiteite is 'n dobbelsteenspeletjie waartydens die besoeker 'n swemkursus of 'n snorkelkursus kan wen as die getalle op die twee dobbelstene wat gegooi is, opeenvolgende getalle is.

**LET WEL:** Elke dobbelsteen het ses kante.  
Opeenvolgende getalle is getalle wat op mekaar volg.

Skryf kode om die volgende te doen:

- Genereer twee ewekansige getalle om die gooi van elke dobbelsteen te simuleer en vertoon die getalle in die afvoerarea.
- Maak knoppie **Vraag 1\_5\_2** en die twee radioknoppies aktief as twee opeenvolgende getalle gegenereer is.

Voorbeeld van afvoer as die getalle wat genereer is, nie opeenvolgend is nie:

Voorbeeld van afvoer as die getalle wat genereer is, opeenvolgend is:

(8)

### Knoppie [Vraag 1\_5\_2]

Kies een van die kursusse as 'n prys en skryf kode om die volgende te doen:

- Gebruik 'n dialoogblokkie om die besoeker se viersyfer-kaartjienommer in te sleutel. Hierdie nommer kan enige viersyfer-getal wees.
- Onttrek ('extract') die stelseldatum in enige formaat.
- Skep 'n verwysingsnommer in die volgende formaat:

**<Kaartjienommer>#<datum>#<eerste twee letters van die geselekteerde kursus as hoofletters>**

Voorbeeld van afvoer as die kaartjienommer wat ingesleutel is, 1028 is, die datum 19 Oktober 2016 is en die kursus wat geselekteer is, swem is:

(8)

- Sleutel jou eksamennommer as kommentaar in die eerste reël van die programlêer in.
- Stoor jou program.
- Maak 'n drukstuk van die kode as dit vereis word.

**TOTAAL AFDELING A: 50**

**AFDELING B****VRAAG 2: OBJEK-GEÛRIËNTEERDE PROGRAMMERING**

Aqua Wonderland bied 'n spesiale vyfdag- opvoedkundige program by die akwarium aan. Skole kan met die administrateurs by die park reël om leerders op 'n uitstappie na die akwarium te neem op enige een van die spesifieke dae waarop die program aangebied word. As 'n skool vra om die park op 'n spesifieke datum te besoek, sal die administrateurs bevestig of die skool dit kan bywoon, afhangend van die plek wat beskikbaar is om die groep leerders te akkommodeer.

Doen die volgende:

<b>Delphi-programmeerders</b>	<b>Java-programmeerders</b>
<ul style="list-style-type: none"> <li>• Maak die onvoltooide program in die <b>Vraag2-lêergids</b> oop.</li> <li>• Maak die onvoltooide objekklas <b>Uitstappie_U.pas</b> oop.</li> <li>• Sleutel jou eksamennommer as kommentaar in die eerste reël van beide lêer <b>Vraag2_U.pas</b> en <b>Uitstappie_U.pas</b> in.</li> </ul>	<ul style="list-style-type: none"> <li>• Maak die onvoltooide program in die <b>Vraag2-lêergids</b> oop.</li> <li>• Maak die onvoltooide objekklas <b>Uitstappie.java</b> oop.</li> <li>• Sleutel jou eksamennommer as kommentaar in die eerste reël van beide objekklas <b>Vraag2.java</b> en <b>Uitstappie.java</b> in.</li> </ul>

- Kompileer en voer die program uit. Die program het tans geen funksionaliteit nie. 'n Paneel met die naam **pnlBeskikbaarheid**, wat weggesteek is, sal nie sigbaar wees as die program uitgevoer word nie en sal in VRAAG 2.2.2 en VRAAG 2.2.3 gebruik word.

Voorbeeld van gebruikerskoppelvlak:

- Voltooi die kode vir hierdie program soos in VRAAG 2.1 en VRAAG 2.2 hieronder gespesifiseer.

2.1 Die gegewe onvoltooide objekklas (**TUitstappie/Uitstappie**) bevat die verklaring van vier attribute en metodes wat 'n **Uitstappie**-objek beskryf.

Die attribute vir die **Uitstappie**-objek is soos volg:

NAME VAN ATTRIBUTE		BESKRYWING
Delphi	Java	
fSkoolnaam	skoolnaam	Die naam van die skool
fBesoekDatum	besoekDatum	Die datum waarop die skool die akwarium wil besoek in die formaat JJJJ-MM-DD
fGroepGrootte	groepGrootte	Die getal leerders in die groep
fToerGids	toerGids	'n Boolese attribuut met die waarde van 'true' as die skool 'n toergids benodig of 'false' as die skool nie 'n toergids benodig nie

Voltooi die kode in die gegewe **Uitstappie**-objekklas (**TUitstappie/Uitstappie**) soos in VRAAG 2.1.1 tot VRAAG 2.1.5 hieronder beskryf word.

2.1.1 Skryf 'n wysigingsmetode ('mutator method') met die naam **setBesoekDatum** wat 'n datum as parameter sal ontvang en die huidige datum waarop die akwarium besoek gaan word, vervang met die datum wat ontvang is. (2)

2.1.2 Skryf 'n metode met die naam **benodigToerGids** wat die woord 'Ja' sal terugstuur as 'n toergids benodig word, of 'Nee' as 'n toergids nie benodig word nie. (4)

2.1.3 'n Maksimum van 500 besoekers word elke dag toegelaat om die program by die akwarium by te woon. Die skool sal nie toegelaat word om die program by te woon as daar nie op die spesifieke datum waarvoor gevra is, plek beskikbaar is nie.

Skryf 'n metode met die naam **isBevestig** wat die totale getal besoekers wat reeds die program op die datum waarvoor gevra is, bywoon, as 'n parameter sal ontvang. Die metode moet die parameterwaarde gebruik om te bepaal of die maksimum getal van 500 besoekers per dag oorskry gaan word, of nie, as hierdie skool toegelaat word om die program op die datum waarvoor gevra is, te besoek. Stuur 'n Boolese waarde van 'true' terug as die skool toegelaat word om die program op grond van beskikbare plek by te woon, of 'false' indien nie. (5)

2.1.4 Die park wil skole aanmoedig om die akwariumprogram by te woon deur aan een leerder, vir elke tien leerders in die groep, gratis toegang te gee.

Skryf 'n metode met die naam **berekenBedrag** wat die koste per persoon en die koste van 'n toergids as parameters sal ontvang en die totale bedrag wat betaalbaar is, sal bereken en terugstuur. (7)

- 2.1.5 Voltooi die **toString**-metode om die inligting oor die uitstappie in die volgende formaat terug te stuur:

```
Skool se naam: <skoolnaam>
Datum van besoek: <datum van besoek>
Getal leerders: <grootte van die groep>
Toergids: <Ja> of <Nee>
```

Voorbeeld van afvoer:

```
Skool se naam: Forest Manor High
Datum van besoek: 2016-11-17
Getal leerders: 145
Toergids: Ja
```

(4)

- 2.2 'n Onvoltooide klas **Vraag2\_U/Vraag2** is voorsien. Besonderhede van die skool wat aansoek doen om op 'n uitstappie te gaan om die akwarium te besoek, moet deur die gebruiker ingesleutel word. Die program moet bepaal of die groep op die datum waarvoor gevra is, geakkommodeer kan word en alternatiewe datums voorsien as die groep nie op die datum waarvoor gevra is, geakkommodeer kan word nie.

'n Tekslêer met die naam **DataV2.txt** bevat inligting oor al die skole wat aanvaar is om die akwarium se opvoedkundige program by te woon.

Die tekslêer sal in VRAAG 2.2.2 gebruik word.

Die formaat van elke reël in die tekslêer is soos volg:

```
<Skool se naam>#<Datum van besoek>#<Getal leerders>
```

Die eerste vier reëls van die tekslêer lyk soos volg:

```
Senzani Primary#2016-11-17#250
Florence High#2016-11-14#235
Golden Palm Secondary#2016-11-17#245
Forester Primary#2016-11-15#150
```

### 2.2.1 Knoppie [2.2.1 – Instansieer objek]

Die gebruiker moet die naam van die skool insleutel, die datum waarop hulle die akwarium wil besoek uit die lysblokkie selekteer en die getal leerders in die groep insleutel.

'n **Uitstappie**-objek met die naam **objUitstappie** is globaal verklaar. Skryf kode om die data wat ingesleutel is, te gebruik om 'n nuwe **Uitstappie**-objek te instansieer.

Vertoon 'n boodskap deur 'n dialoogblokkie te gebruik om aan te dui dat die objek suksesvol geïnstansieer is.

(7)

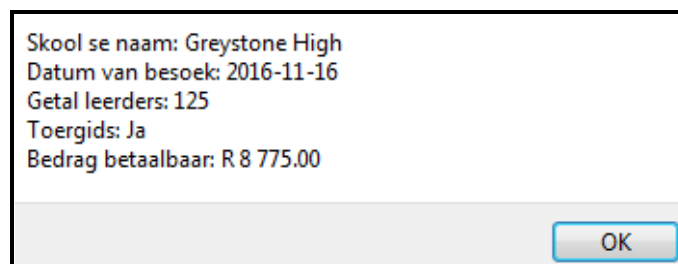
### 2.2.2 Knoppie [2.2.2 – Bevestig beskikbaarheid]

- (a) Skryf 'n metode met die naam **bepaalDagTotaal** wat die datum wat die skool gekies het om die akwarium te besoek, as parameter ontvang. Gebruik die data wat in die **DataV2.txt**-tekst lêer gestoor is, om die totale getal besoekers wat die program bywoon op die datum wat deur die metode ontvang is, te bepaal en terug te stuur. (12)
- (b) Gebruik die dagtotaal se waarde wat deur die **bepaalDagTotaal**-metode bepaal is as 'n parameter en roep die **isBevestig**-metode om te bepaal of die skool ingesluit kan word by die akwarium se program op die geselekteerde datum, of nie.

As die skool toegelaat word om die akwarium op die geselekteerde datum te besoek, gebruik die **berekenBedrag**-metode om die bedrag te bereken wat betaal moet word.

**LET WEL:** Die koste per persoon om die park te besoek en die koste om 'n toergids te huur is verklaar as konstante veranderlikes as deel van die kode wat voorsien is.

Vertoon die skool se besonderhede en die bedrag wat betaal moet word deur 'n boodskap-dialoogblokkie te gebruik, soos in die voorbeeld hieronder getoon word.



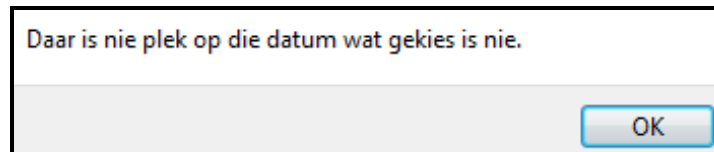
As die skool nie die akwarium op die geselekteerde datum kan besoek nie, doen die volgende:

- Vertoon 'n boodskap in 'n dialoogblokkie wat aandui dat daar nie plek beskikbaar is nie.
- Toon die paneel met die naam **pnlBeskikbaarheid** wat tans weggesteek is.
- Toets al die datums in die lysblokkie vir beskikbaarheid deur die **bepaalDagTotaal**- en die **isBevestig**-metode te gebruik. Vul die kombinasieblokkie ('combo box') in die **pnlBeskikbaarheid**-paneel met die datums waarop die skool die akwarium kan besoek.

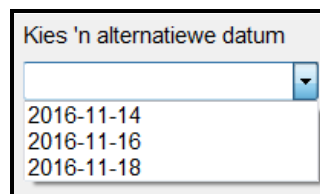
Gebruik die volgende skoolbesonderhede om die program te toets:

Skool se naam: Hovener Secondary  
 Datum van besoek: 2016-11-15  
 Getal leerders: 150  
 Toergids: Ja

Voorbeeld van afvoer as die skool nie op die geselekteerde datum van die besoek ingesluit kan word nie:



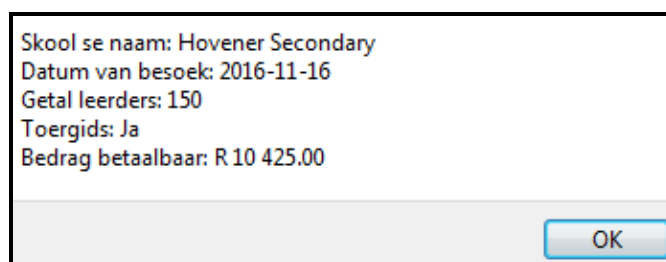
Die kombinasieblokkie ('combo box') op die paneel **pnBesikbaarheid** wat gevul is met al die datums waarop die skool op grond van beskikbare plek die akwarium kan besoek:



(16)

### 2.2.3 Knoppie [2.2.3 – Bevestig nuwe datum]

As alternatiewe datums in die kombinasieblokkie ('combo box') vertoon word, moet die gebruiker 'n datum kies. Die program moet die datum van die objek op die geselekteerde datum stel en die skool se inligting saam met die bedrag betaalbaar in 'n dialoogblokkie vertoon, soos in die voorbeeld hieronder getoon.



As daar nie alternatiewe datums in die kombinasieblokkie ('combo box') vertoon word nie, vertoon 'n boodskap wat aandui dat die skool se versoek om op die uitstappie te gaan, onsuksesvol was.

(5)

- Sleutel jou eksamennommer as kommentaar in die eerste reël van die klas en die vorm in.
- Stoor al die lêers.
- Druk die kode van beide die klas en die vorm (Delphi)/albei klasse(Java) indien dit vereis word.

**AFDELING C****VRAAG 3: PROBLEEMOPLOSSINGPROGRAMMERING****SCENARIO**

Inligtingsentrums, wat 'terminale' genoem word, is op drie verskillende punte langs die roetes in die park opgerig. Die bestuur benodig 'n sagtewareprogram wat inligting aan besoekers kan verskaf om vas te stel waar die verskillende aktiwiteite en fasiliteite in Aqua Wonderland is en om bewus te wees van instandhoudingsaktiwiteite in die park.

Doen die volgende:

- Kompileer en voer die program in die **Vraag3**-lêergids uit. Die program het tans geen funksionaliteit nie.
- Voltooi die kode vir elke vraag soos in VRAAG 3.1 tot VRAAG 3.4 beskryf is.

**GGK ('GUI') wat voorsien is:**

Die GGK ('GUI') wat voorsien is, stel 'n selfhelp-koppelvlak voor wat besoekers help om by verskeie aktiwiteite en fasiliteite in die park uit te kom.

**Data wat voorsien is:**

Jy word voorsien van twee parallelle skikkings en een tweedimensionele skikking.

**arrAktiwiteite** is 'n eendimensionele skikking wat die name van aktiwiteite en fasiliteite in die park bevat. Die data wat in hierdie skikking gestoor word, is soos volg:

Waterpark, Akwarium, See, Restaurante, Inkopies, Duik, Hulptonbank,  
Pikkewynpark, Haaitenk, Dolfynvertonings

'n Ooreenstemmende parallelle skikking met die naam **arrKodes** bevat letters van die alfabet wat elk die ooreenstemmende aktiwiteit/fasiliteit voorstel wat in die **arrAktiwiteite**-skikking beskryf word.

Die **arrKodes**-skikking bevat die volgende elemente:

W, A, S, R, X, D, H, P, T, L

Die eerste element (letter W) in die **arrKodes**-skikking verteenwoordig die eerste element ('Waterpark') in die **arrAktiwiteite**-skikking, die tweede element (letter A) in die **arrKodes**-skikking verteenwoordig die tweede element ('Akwarium') in die **arrAktiwiteite**-skikking, en so aan.

**arrAktKodes** is 'n tweedimensionele skikking wat 'n kombinasie van kodes bevat wat aktiwiteite en fasiliteite verteenwoordig wat vanaf 'n spesifieke terminaal toeganklik is as die besoeker in 'n spesifieke rigting beweeg. Die kodes wat in hierdie skikking voorkom, is soos volg:

	Noord	Suid	Oos	Wes
Terminaal 1	DXWAT	HRDST	STWLP	RDT
Terminaal 2	SWA	SRXD	LWXH	SHA
Terminaal 3	WLSR	AT	DATX	HW

**LET WEL:** Die opskrifte van die rye en kolomme word nie as deel van die tweedimensionele skikking voorsien nie.

Voorbeeld:

Die aktiwiteitskode wat van toepassing is as 'n besoeker vanaf Terminaal 1 in 'n noordelike rigting loop, is DXWAT.

Deur die inhoud van die **arrKodes**- en **arrAktiwiteite**-skikking te gebruik, kan daar vasgestel word dat die aktiwiteite en fasiliteite waarna die kode DXWAT verwys, Duik, Inkopies, Waterpark, Akwarium en Haaitenk is.

**LET WEL:**

- Jy mag NIE die data wat voorsien is, met die hand verander nie. Kode moet geskryf word om die data wat voorsien is, volgens die vereistes te manipuleer.
- Die gebruik van goeie programmeringstegnieke en modulêre ontwerp moet in die ontwerp en kodering van jou oplossing toegepas word.

### 3.1 Knoppie [3.1 – Aktiwiteits-/Fasiliteitskodes vir alle terminale en rigtings]

Die program moet die inhoud van die tweedimensionele skikking **arrAktKodes** netjies in rye en kolomme vertoon. Vertoon die rigtings as kolomopskrifte en die terminale as die rye se byskrifte.

Voorbeeld van afvoer:

	Noord	Suid	Oos	Wes
Terminaal 1	DXWAT	HRDST	STWLP	RDT
Terminaal 2	SWA	SRXD	LWXH	SHA
Terminaal 3	WLSR	AT	DATX	HW

(9)

### 3.2 Knoppie [3.2 – Aktiwiteite/Fasiliteite vanaf 'n terminaal en rigting wat gekies is]

Die knoppies wat prentjies bevat, moet gebruik word om 'n terminaal en 'n rigting te kies. Kode word voorsien om die terminaal en rigting wat gekies is, aan veranderlikes toe te ken. Die program moet dan die skikkings wat voorsien is, gebruik om al die beskikbare aktiwiteite en fasiliteite op die roete wat gekies is, te identifiseer. Vertoon die terminaal en rigting wat gekies is, as 'n opskrif en 'n lys van aktiwiteite en fasiliteite op die roete wat gekies is.

Voorbeeld van afvoer as Terminaal 2 en Suid gekies is:

```
Terminaal 2, Suid
See
Restaurante
Inkopies
Duik
```

(11)

### 3.3 Knoppie [3.3 – Toegangsroetes na aktiwiteit/fasiliteit wat gekies is]

Sodra die gebruiker 'n spesifieke aktiwiteit/fasiliteit uit die kombinasieblokkie ('combo box') wat voorsien is, gekies het, moet die gebruiker al die toegangsroetes na daardie spesifieke aktiwiteit/fasiliteit kan sien. Vertoon die nommer van die terminaal en die rigting van elke toegangsroete om die aktiwiteit/fasiliteit wat gekies is, te kan bereik.

Die totale getal toegangsroetes wat lei na die aktiwiteit/fasiliteit wat gekies is, moet bepaal en vertoon word.

Voorbeeld van afvoer as Akwarium uit die kombinasieblokkie ('combo box') gekies is:

```
Toegangsroetes na Akwarium
Terminaal 1, Noord
Terminaal 2, Noord
Terminaal 2, Wes
Terminaal 3, Suid
Terminaal 3, Oos
Getal toegangsroetes: 5
```

(10)

### 3.4 Knoppie [3.4 – Instandhouding van aktiwiteit/fasiliteit wat gekies is]

Die area waar aktiwiteite plaasvind of waar fasiliteite voorsien word, kan soms as gevolg van instandhouding gesluit wees. Die gebruiker moet 'n aktiwiteit/fasiliteit kies waar instandhouding gedoen moet word uit die kombinasieblokkie ('combo box') wat voorsien is. Die program moet alle verwysings na die aktiwiteit/fasiliteit wat gekies is, uit die tweedimensionele skikking verwyder en 'n geskikte boodskap in 'n dialoogblokkie vertoon wat aandui dat die inligting opgedateer is. Die opgedateerde inhoud van die tweedimensionele skikking moet in die afvoerarea vertoon word.

Voorbeeld van afvoer as Duik uit die kombinasieblokkie ('combo box') gekies is:



Voorbeeld van afvoer nadat die letter D uit die tweedimensionele skikking **arrAktKodes** verwyder is as gevolg van instandhouding wat aan die duikfasiliteit gedoen moet word:

Opgedateerde inligting:				
	Noord	Suid	Oos	Wes
Terminaal 1	XWAT	HRST	STWLP	RT
Terminaal 2	SWA	SRX	LWXH	SHA
Terminaal 3	WLSR	AT	ATX	HW

(8)

- Sleutel jou eksamennommer as kommentaar in die eerste reël van die programlêer in.
- Stoor jou program.
- Maak 'n drukstuk van die kode as dit vereis word.

**TOTAAL AFDELING C: 38**  
**GROOTTOTAAL: 150**



# basic education

---

Department:  
Basic Education  
**REPUBLIC OF SOUTH AFRICA**

**NASIONALE  
SENIOR SERTIFIKAAT**

**GRAAD 12**

**INLIGTINGSTEGNOLOGIE V1**

**NOVEMBER 2016**

**MEMORANDUM**

**PUNTE: 150**

**Hierdie vraestel bestaan uit 32 bladsye.**

**ALGEMENE INLIGTING:**

- Hierdie nasienriglyne moet as die basis vir die nasiensessie gebruik word. Dit is voorberei om deur nasieners gebruik te word. Daar word na alle nasieners verwag om 'n deeglike standaardiseringsvergadering by te woon om seker te maak dat die riglyne konsekwent geïnterpreteer en tydens die nasien van die kandidate se werk toegepas word.
- Let op dat leerders wat 'n alternatiewe korrekte oplossing as wat as voorbeeld van 'n oplossing in die nasienriglyne gegee word verskaf, volle krediet vir die relevante oplossing moet kry tensy die spesifieke instruksies in die vraestel nie gevolg is nie of die vereistes van die vraag nie nagekom is nie.
- **Bylaag A, B en C** (bladsy 3-10) sluit die nasienrubriek vir elke om te gebruik vir enigeen van die twee programmeringstale in.
- **Bylaag D, E en F** (bladsy 11-19) bevat voorbeelde in programmeringskode in Java van oplossings vir VRAAG 1 tot VRAAG 3.
- **Bylaag G, H en I** (bladsy 20-32) bevat voorbeelde in programmeringskode in Delphi van oplossings vir VRAAG 1 tot VRAAG 3.
- Kopieë van **Bylaag A, B en C** (bladsy 3-10) moet vir elke leerder gemaak word en tydens die nasiensessie voltooi word.

**BYLAAG A:****AFDELING A:****VRAAG 1: NASIENRUBRIEK - ALGEMENE PROGRAMMERINGSVAARDIGHEDE**

SENTRUMNOMMER:		EKSAMENNOMMER:	
VRAAG	BESKRYWING	MAKS. PUNTE	LEERDER-PUNT
	'n Leerder moet slegs een keer gepenaliseer word indien dieselfde fout herhaaldelik voorkom.		
1.1	Onttrek lengte uit teksblokkie ✓, skakel om na reële getal ✓ Onttrek breedte uit teksblokkie en skakel om na getal } Onttrek hoogte uit teksblokkie en skakel om na getal } ✓ <i>Bereken volume deur van die formule gebruik te maak:</i> $\text{Volume} = \text{lengte} * \text{breedte} * \text{hoogte} \checkmark$ Vertoon die volume in die teksblokkie ✓	5	
1.2	<b>Knoppie – [Vraag 1_2]</b>  $\text{Volume} = \text{volume} * 1000$ as $\text{volume} \leq 500 \checkmark$ <i>Bereken koste deur formule te gebruik</i> $\text{Koste} = \text{Volume} * 0.25 \checkmark$ anders as $\text{volume} \leq 800 \checkmark$ <i>Bereken koste deur formule te gebruik</i> $\text{Koste} = 500 * 0.25 \checkmark + (\text{volume} - 500) * 0.35 \checkmark$ anders ✓ <i>Bereken koste deur formule te gebruik</i> $\text{Koste} = 500 * 0.25 + 300 * 0.35 \checkmark + (\text{volume} - 800) * 0.45 \checkmark$ Vertoon die koste in die teksblokkie ✓  Konsepte: <ul style="list-style-type: none"> <li>• 2 punte: As <math>\text{volume} \leq 500</math> en berekening</li> <li>• 3 punte: As <math>\text{volume}</math> meer as 500 is en kleiner of = 800 en berekening</li> <li>• 1 punt: anders</li> <li>• 2 punte: as <math>\text{volume} &gt; 800</math> en berekening</li> <li>• 1 punt: vertoon</li> </ul>	9	

1.3	<p><b>Knoppie - [Vraag 1_3]</b></p> <p>Kry die leeftyd in maande uit die teksblokkie as 'n getal ✓</p> <p><i>Bereken die getal maande deur mod of enige ander metode maande = leeftyd modulus 12 ✓</i></p> <p><i>Bereken die getal jaar deur die korrekte formule te gebruik</i>  <i>jare = leeftyd deel deur 12 ✓ (verwyder die waarde na die desimale punt) ✓</i></p> <p>Vertoon jare ✓ en maande ✓</p>	6	
1.4	<p><b>Knoppie - [Vraag 1_4]</b></p> <p>Onttrek opstellingskoste uit teksblokkie, skakel om na getal ✓</p> <p>Onttrek die inkomste van eerste jaar uit teksblokkie, skakel om na getal ✓</p> <p>Vertoon opskrifte in afvoerarea ✓</p> <p>Inisialiseer Jaarnommer-veranderlike na 1 ✓</p> <p>Gebruik lus om oprigKoste te toets <math>&gt; 0</math> ✓</p> <p style="padding-left: 20px;"><math>\text{oprigKoste} = \text{oprigKoste} - \text{jaarlikse inkomste}</math> ✓</p> <p style="padding-left: 20px;">as <math>\text{oprigKoste} &gt; 0</math> ✓</p> <p style="padding-left: 40px;">vertoon jaarNommer, jaarlikse inkomste, oprigKoste ✓</p> <p style="padding-left: 20px;">anders</p> <p style="padding-left: 40px;">vertoon jaarNommer, jaarlikse inkomste, "Afbetaal" ✓</p> <p>Vermeerder jaarNommer met 1 ✓</p> <p>Vermeerder met 10% ✓ jaarlikse inkomste ✓</p> <p>Alle geldwaardes moet na geldeenheid geformateer wees ✓ met twee desimale plekke ✓</p> <p><i>Alternatiewe oplossing:</i></p> <p>Onttrek opstellingskoste uit teksblokkie, skakel om na getal (1 punt)</p> <p>Onttrek die inkomste van eerste jaar uit teksblokkie, skakel om na getal (1 punt)</p> <p>Vertoon opskrifte in afvoerarea (1 punt)</p> <p>Inisialiseer Jaarnommer-veranderlike na 1 (1 punt)</p> <p>Gebruik lus om oprigKoste te toets <math>&gt; \text{inkomste}</math> (2 punt)</p> <p style="padding-left: 20px;"><math>\text{oprigKoste} = \text{oprigKoste} - \text{jaarlikse inkomste}</math> (1 punt)</p> <p style="padding-left: 20px;">as <math>\text{oprigKoste} &gt; \text{inkomste}</math> (1 punt)</p> <p style="padding-left: 40px;">vertoon jaarNommer, jaarlikse inkomste, oprigKoste (1 punt)</p> <p style="padding-left: 20px;">Vermeerder jaarNommer met 1 (1 punt)</p> <p style="padding-left: 20px;">Vermeerder jaarlikse inkomste (1 punt) met 10% (1 punt)</p> <p>vertoon jaarNommer, jaarlikse inkomste, "Afbetaal" (1 punt)</p>	14	

1.5	<p><b>Knoppie - [Vraag 1_5_1]</b></p> <p>Genereer 'n willekeurige waarde ✓ vir dobbelsteen 1 in regte reeks ✓          Genereer 'n willekeurige waarde vir dobbelsteen 2 ✓          Vertoon dobbelsteen 1 byskrif en waarde in afvoerarea }          Vertoon dobbelsteen 2 byskrif en waarde in afvoerarea } ✓</p> <p><i>Toets of getalle vir dobbelsteen 1 en 2 opeenvolgend is</i></p> <p>Toets as dobbelsteen 1 – dobbelsteen 2 = 1 ✓          Toets OF as dobbelsteen 2 – dobbelsteen 1 = 1 ✓</p> <p><i>Alternatiewe oplossing:</i>          As <math>\text{abs}(\text{doppelsteen1} - \text{doppelsteen2}) = 1</math> (2 punte)</p> <p>Maak radioknoppies/radiogroep aktief ✓          Maak Vraag 1_5_2-knoppie aktief ✓</p>	8	
	<p><b>Knoppie - [Vraag 1_5_2]</b></p> <p>Gebruik 'n dialoogblokkie om die kaartjienommer in te sleutel ✓          Verkry die stelseldatum ✓✓</p> <p>Selekteer die opsie van die radio button ✓</p> <p>Stel die verwysingsnommer saam deur:              die kaartjienommer en datum ✓              die eerste twee letters van die kursus in hoofletters ✓              # in korrekte plekke ✓</p> <p>Vertoon die verwysingsnommer in die afvoerarea ✓</p>	8	
	<b>TOTAAL:</b>	<b>50</b>	

**BYLAAG B:****AFDELING B:****VRAAG 2: NASIENRUBRIEK - OBJEK-GEÛRIENTEERDE PROGRAMMERING**

SENTRUMNOMMER:		EKSAMENNOMMER:	
VRAAG	BESKRYWING	MAKS. PUNTE	LEERDER-PUNT
2.1.1	<b>Wysingingsmetode vir setDatumVanBesoek:</b> Metode se definisie met datum parameter ✓ Ken die parameterwaarde toe aan die attribuut ✓	2	
2.1.2	<b>vereisToerGids-metode:</b> Terugstuur-datatipe String ✓ Toets of toerGids = true ✓ Stuur terug ✓ "Ja" ✓ ( <i>Regte woord</i> ) anders Stuur "Nee" terug	4	
2.1.3	<b>isBevestig-metode:</b> Metode definisie met parameter en boolean terugstuur-datatipe ✓ Toets of dagTotaal + groepGrootte ✓ ≤ 500 ✓ Stuur true terug ✓ else Stuur false terug ✓	5	
2.1.4	<b>berekenBedrag-metode</b> Metode se definisie met 2 korrekte parameters ✓  <i>Bereken hoeveel leerdere gratis toegang het:</i> aantGratis = groepGrootte gedeel deur 10 om heelgetal antwoord te kry ✓  <i>Bereken koste:</i> bedrag = (groepGrootte – aantGratis) ✓ *kostePerPersoon ✓  Toets of toergids vereis word ✓ bedrag = bedrag + kosteToerGids ✓ Stuur bedrag terug ✓	7	
2.1.5	<b>toString-metode:</b> Byskrifte elke item op nuwe reël ✓ Al die regte attributes/getMethods ✓ Roep metode om toergids te bevestig ✓ Stuur die saamgestelde string terug ✓	4	

2.2.1	<p><b>Knoppie - [2.2.1 Instansieer objek]</b></p> <p>Verkry naam van skool uit teksblokkie ✓          Verkry die datum uit lysblokkie ✓          Verkry die grootte van die groep uit teksblokkie en skakel om na heelgetal ✓          As kontroleblokkie afgemerkt is ✓              Stel die toerGids op true anders              Stel die toerGids op false ✓</p> <p>Instansieer objek met korrekte parameters ✓          Vertoon boodskap om aan te dui dat objek geinstasieer is ✓</p>	7	
2.2.2(a)	<p><b>Knoppie – [2.2.2 – Bevestig beskikbaarheid]</b></p> <p><b>bepaalDagTotaal</b>-metode met datum parameter ✓</p> <p>Stel dagTotaal op 0 ✓</p> <p><i>Lees en skei inligting in tekslêer</i>  <i>{Delphi: AssignFile, Reset</i>  <i>Java: Skep objek om uit leer te lees} ✓✓</i></p> <p>Lus deur die lêer ✓              Lees 'n reël op 'n slag ✓              Vind die posisie van die datum ✓              Onttrek die datum van besoek uit die reël ✓              Verkry die grootte van die groep uit die reël ✓</p> <p>Toets of datum uit reël = datum ontvang as parameter ✓              Vermeerder dagTotaal met groepGrootte ✓</p> <p>CloseFile(Tekslêer);          Stuur die finale dagTotaal terug ✓</p>	12	

2.2.2(b)	<p>Maak gebruik van die bepaalDagTotaal-metode om die dagTotaal waarde te verkry ✓</p> <p><i>Stuur die dagTotaal as argument na die isBevestig-metode van die objekklas</i></p> <p>Toets of die isBevestig-metode true terugstuur ✓  Roep die berekenBedrag-metode ✓  deur 2 konstante waardes as argumente te stuur ✓  om bedrag te bereken</p> <p>Vertoon die string wat deur die toString-metode teruggestuur is ✓ en die bedrag wat deur die berekenBedrag-metode teruggestuur is ✓</p> <p>Anders ✓  Vertoon 'n boodskap wat geen beskikbaarheid aandui ✓  Vertoon die paneel met die naam pnlBesikbaarheid ✓  Lus deur die lysblokkie ✓  Onttrek datum uit die lysblokkie ✓  Roep bepaalDagTotaal-metode  Deur datum wat onttrek is as argument te gebruik ✓  Stuur die dagTotaal ✓ na die isBevestig-metode ✓  As die isBevestig-metode true terugstuur ✓  Voeg die datum by items in kombinasieblokkie ✓</p>	16	
2.2.3	<p><b>Knoppie – [2.2.3 – Bevestig nuwe datum]</b></p> <p>Toets of daar enige items in die kombinasieblokkie is ✓  Verkry die geselekteerde datum uit die kombinasieblokkie ✓  Roep die setDatumVanBesoeke-metode met hierdie datum as argument ✓  Roep die toString-metode en die berekenBedrag-metode om te vertoon ✓</p> <p>Anders  Vertoon 'n gepaste boodskap wat aandui dat die aansoek om op uitstappie te gaan onsuksesvol was ✓</p>	5	
	<b>TOTAAL:</b>	<b>62</b>	

**BYLAAG C:****AFDELING C:****VRAAG 3: NASIENRUBRIEK - PROBLEEMOPLOSSING**

SENTRUMNOMMER:		EKSAMENNOMMER:	
VRAAG	BESKRYWING	MAKS. PUNTE	LEERDER-PUNT
3.1	<p><b>Knoppie [3.1 – Aktiwiteits-/Fasiliteitskodes vir alle terminale en rigtings]</b></p> <p>Vertoon kolomopskrifte ✓</p> <p>Buitenste lus om rye te beheer ✓            Voeg terminaal se nommer by reël ✓            Binneste lus om kolomme te beheer ✓            Vind korrekte data-element uit tweeD-skikking ✓            Voeg data by reël ✓            Eindig binneste lus            Vertoon die reël in die afvoerarea op 'n nuwe lyn ✓            Eindig buitenste lus</p> <p>Vertoon in netjiese kolomme ✓</p>	9	
	<p>Gebruik 'n datastruktuur of enige ander metode om die rigtings Noord, Suid, Oos en Wes te vind ✓</p>		
3.2	<p><b>Knoppie [3.2 – Aktiwiteite/Fasiliteite vanaf 'n terminaal en rigting wat gekies is]</b></p> <p>Vertoon die terminaal se nommer ✓ en die rigting in die afvoerarea ✓</p> <p>Verkry die kode uit die tweeD-skikking vir hierdie terminaal se nommer ✓ en hierdie rigting ✓</p> <p>Lus deur die lengte van die kode wat onttrek is ✓            Lus deur die arrKodes-skikking ✓            Vergelyk elke letter in die kode ✓ teen die waardes in die arrKodes-skikking is ✓            Vertoon die aktiwiteit uit die arrAktiwiteite-skikking ✓ by dieselfde indeks ✓            as in die arrKodes-skikking ✓</p>	11	

3.3	<p><b>Knoppie [3.3 – Toegangsroetes na aktiwiteit/fasiliteit wat gekies is]</b></p> <p>Verkry indeks van die geselekteerde aktiwiteit/fasiliteit in die kombinasie lys ✓          Stel teller op 0 ✓          Vertoon opskrif ✓          Vind die kode van die geselekteerde aktiwiteit ✓</p> <p>Buitenste lus vir rye ✓          Binneste lus vir kolomme ✓          Toets of die kode deel is van die kode in die tweeD-Skikking ✓              Vertoon die terminaal se nommer en die rigting ✓              Vermeerder die teller met 1 ✓</p> <p>Vertoon die byskrif vir die aantal aktiwiteite en waarde die van die teller ✓</p>	10	
3.4	<p><b>Knoppie [3.4 – Instandhouding van aktiwiteit/fasiliteit wat gekies is]</b></p> <p>Verkry indeks van die geselekteerde aktiwiteit/fasiliteit in die kombinasie lys ✓</p> <p>Buitenste lus vir rye ✓          Binneste lus vir kolomme ✓          Toets of die kode van die geselekteerde aktiwiteit deel is van die kode in die tweeD-skikking ✓              Verwyder die kodeletter uit die tweeD-skikking ✓              By die regte ry en kolom ✓</p> <p>Vertoon 'n boodskap wat aandui dat die toegangsroetes tot die geselekteerde aktiwiteit/fasiliteit gesluit is ✓          Roep 'n metode of 'n knoppie en vertoon die opgedateerde tweeD-skikking met opskrifte ✓</p>	8	
<b>TOTAAL</b>		<b>38</b>	

**OPSOMMING VAN LEERDER PUNTE:**

SENTRUMNOMMER:	EKSAMENNOMMER:			
	AFDELING A	AFDELING B	AFDELING C	
	VRAAG 1	VRAAG 2	VRAAG 3	GROOT-TOTAAL
<b>MAKS. PUNTE</b>	50	62	38	150
<b>LEERDER PUNTE</b>				

**BYLAAG D: OPLOSSING VIR VRAAG 1: JAVA**

```
// Gegewe kode
SimpleDateFormat sdf = new SimpleDateFormat("dd/MM/yyyy");

// Globale veranderlikes
DecimalFormat df1 = new DecimalFormat("0.00"); // Vr1.1
DecimalFormat df = new DecimalFormat("R0.00"); // Vr1.2
double volume = 0; // Vr1.1 & Vr1.2

=====
// Vraag 1.1
=====
private void btnVraag11ActionPerformed(java.awt.event.ActionEvent evt) {
    double lengte, breedte, hoogte;
    lengte = Double.parseDouble(txfLengte.getText());
    breedte = Double.parseDouble(txfBreedte.getText());
    hoogte = Double.parseDouble(txfHoogte.getText());
    volume = lengte * breedte * hoogte;
    txfVolume.setText("" + df1.format(volume));
}
=====
// Vraag 1.2
=====
private void btnVraag12ActionPerformed(java.awt.event.ActionEvent evt) {
    double koste = 0;
    if (volume <= 500) {
        koste = volume * 0.25;
    } else if (volume <= 800) {
        koste = (500 * 0.25) + (volume - 500) * 0.35;
    } else {
        koste = (500 * 0.25) + (300 * 0.35) + (volume - 800) * 0.45;
    }
    txfKoste.setText("" + df.format(koste));
}
=====
// Vraag 1.3
=====
private void btnVraag13ActionPerformed(java.awt.event.ActionEvent evt) {
    int leeftydInMaande = Integer.parseInt(txfLeeftydInMaande.getText());
    int maande = leeftydInMaande % 12;
    int jare = (leeftydInMaande - maande) / 12;
    txfJareEnMaande.setText(jare + " jaar en " + maande + " maande");
}
=====
// Vraag 1.4
=====
private void btnVraag14ActionPerformed(java.awt.event.ActionEvent evt) {
    txaVraag14.setText(String.format("%-10s%-14s%-15s\n",
        "Jaar", "Inkomste", "Balans"));
    double oprigKoste = Double.parseDouble(txfOprigKoste.getText());
    double jaarInkomste = Double.parseDouble(txfInkomsteJaar1.getText());
    int jaarNommer = 1;
    while (oprigKoste > 0) {
        oprigKoste = oprigKoste - jaarInkomste;
        if (oprigKoste > 0) {
            txaVraag14.append(String.format("%-10sR%-13.2fR%-13.2f\n",
                jaarNommer, jaarInkomste, oprigKoste));
        }
    }
}
=====
```

```

    }
else {
    txaVraag14.append(String.format("%-10sR%-13.2f%-15s\n",
        jaarNommer, jaarInkomste, "Afbetaal"));
    }
    jaarNommer++;
    jaarInkomste = jaarInkomste * 1.1;
}
}

```

```

=====
// Vraag 1.5.1
=====

```

```

private void btnVraag151ActionPerformed(java.awt.event.ActionEvent evt) {
    int dobbel1 = (int) (Math.random() * 6) + 1;
    int dobbel2 = (int) (Math.random() * 6) + 1;
    txaVraag15.setText("Dobbelsteen 1 = " + dobbel1);
    txaVraag15.append("\nDobbelsteen 2 = " + dobbel2);
    if ((dobbel1 == (dobbel2 + 1)) || (dobbel1 == (dobbel2 - 1))) {
        rgbSnorkel.setEnabled(true);
        rgbSwem.setEnabled(true);
        btnVraag152.setEnabled(true);
    }
    else{
        rgbSnorkel.setEnabled(false);
        rgbSwem.setEnabled(false);
        btnVraag152.setEnabled(false);
    }
}
}

```

```

=====
// Vraag 1.5.2
=====

```

```

private void btnVraag152ActionPerformed(java.awt.event.ActionEvent evt) {
    String kaartjieNom = JOptionPane.showInputDialog("Sleutel jou kaartjie
se nommer in");
    Date vandag = new Date();
    String datum = sdf.format(vandag);
    String verwysingsNom = kaartjieNom + "#" + datum + "#";
    if (rgbSnorkel.isSelected()) {
        verwysingsNom = verwysingsNom +
            rgbSnorkel.getText().substring(0,2);
    }
    else{
        verwysingsNom = verwysingsNom + rgbSwem.getText().substring(0,2);
    }
    txaVraag15.append("\nVerwysingsnommer:\n"+verwysingsNom.toUpperCase());
}
}

```

**BYLAAG E: OPLOSSING VIR VRAAG 2: JAVA****OPLOSSING VIR VRAAG 2: OBJEKKLAS**

```
public class Uitstappie {  
  
=====  
// Kode wat voorsien is  
=====  
private String skoolNaam, besoekDatum;  
private int groepGrootte;  
private boolean toerGids;  
  
public Uitstappie(String skoolNaam, String besoekDatum, int groepGrootte,  
                  boolean toerGids) {  
    this.skoolNaam = skoolNaam;  
    this.besoekDatum = besoekDatum;  
    this.groepGrootte = groepGrootte;  
    this.toerGids = toerGids;  
}  
  
public String getSkoolNaam() {  
    return skoolNaam;  
}  
  
public String getBesoekDatum() {  
    return besoekDatum;  
}  
  
public int getGroepGrootte() {  
    return groepGrootte;  
}  
  
=====  
// Vraag 2.1.1  
=====  
public void setBesoekDatum(String besoekDatum) {  
    this.besoekDatum = besoekDatum;  
}  
  
=====  
// Vraag 2.1.2  
=====  
public String vereisToerGids(){  
    if (toerGids){  
        return "Ja";  
    }  
    else{  
        return "Nee";  
    }  
}  
  
=====  
// Vraag 2.1.3  
=====  
public boolean isBevestig(int dagTotaal){  
    if ((dagTotaal + groepGrootte) <= 500 ) {  
        return true;  
    }  
    else{  
        return false;    }}  
}
```

```
=====
// Vraag 2.1.4
=====
```

```
public double berekenBedrag(double persoonKoste, double toerKoste) {
    int  aantGratis = groepGrootte / 10;
    double bedrag = (groepGrootte - aantGratis) * persoonKoste;
    if (toerGids) {
        bedrag += toerKoste;
    }
    return bedrag;
}
```

```
=====
// Vraag 2.1.5
=====
```

```
public String toString() {
    return "Skool se naam: " + skoolNaam + "\nDatum van besoek: "+
        besoekDatum + "\nGetal leerders: " + groepGrootte + "\nToergids: "
        + vereisToerGids();
}
}
```

**GGK-KLAS: VRAAG2\_OPLOSSING**

```
=====
// Kode wat voorsien is
=====
public class Vraag2 extends javax.swing.JFrame {

public Vraag2() {
    initComponents();
    pnlBeskikbaarheid.setVisible(true);
    setLocationRelativeTo(this);
}

Uitstappie objUitstappie;
final double kostePerPersoon = 75.00;
final double kosteToerGids = 300.00;

=====
// Vraag 2.2.1
=====
    private void btnVraag221ActionPerformed(java.awt.event.ActionEvent evt) {
        String skool = txfSkoolNaam.getText();
        String datum = "" + lstBesoekDatum.getSelectedValue();
        int groepGrootte = Integer.parseInt(txfGroepGrootte.getText().trim());
        boolean toerGids = false;
        if (chkToerGids.isSelected()) {
            toerGids = true;
        }
        objUitstappie = new Uitstappie(skool, datum, groepGrootte, toerGids);
        JOptionPane.showMessageDialog(null, "Uitstappie-objek is
                                           geinstansieer.");
        pnlBeskikbaarheid.setVisible(false);
    }
}

=====
// Vraag 2.2.2 (a)
=====
    public int bepaalDagTotaal(String besoekDatum) {
        int totaal = 0;
        try {
            Scanner sc = new Scanner (new FileReader("DataV2.txt"));
            while (sc.hasNext()) {
                String reel = sc.nextLine();
                String [] arrInfo = reel.split("#");
                String skoolNaam = arrInfo[0];
                String datum = arrInfo[1];
                int groepGrootte = Integer.parseInt(arrInfo[2]);
                if (datum.equalsIgnoreCase(besoekDatum)) {
                    totaal += groepGrootte;
                }
            }
            sc.close();
        } catch (Exception e) {
            JOptionPane.showMessageDialog(null, "Fout:" + e.getMessage());
        }
        return totaal;
    }
}
=====
```

```
=====
// Vraag 2.2.2 (b)
=====
```

```
private void btnVraag222ActionPerformed(java.awt.event.ActionEvent evt) {
    int totaal = bepaalDagTotaal(objUitstappie.getBesoekDatum());
    boolean suksesvol = false;
    if (objUitstappie.isBevestig(totaal)) {
        JOptionPane.showMessageDialog(null,objUitstappie.toString()+"\nBedrag
betaalbaar: "+df.format(objUitstappie.berekenBedrag(kostePerPersoon,
kosteToerGids)));
        suksesvol = true;
    }
    else{
        JOptionPane.showMessageDialog(null,"Daar is nie plek op die datum
wat gekies is nie." );
        pnlBeskikbaarheid.setVisible(true);
        cmbBeskikbareDatums.removeAllItems();
        for (int i = 0; i < 5; i++) {
            lstBesoekDatum.setSelectedIndex(i);
            String datum = "" + lstBesoekDatum.getSelectedValue();
            System.out.println(datum);
            totaal = bepaalDagTotaal(datum);
            if (objUitstappie.isBevestig(totaal)) {
                cmbBeskikbareDatums.addItem("" + datum);
            }//if
        }//for
    }//else
}
=====
```

```
// Vraag 2.2.3
=====
```

```
private void btnVraag223ActionPerformed(java.awt.event.ActionEvent evt) {
    if (cmbBeskikbareDatums.getItemCount() >= 0) {
        objUitstappie.setBesoekDatum("" +
        cmbBeskikbareDatums.getSelectedItem());
        JOptionPane.showMessageDialog(null,objUitstappie.toString()+"\nBedrag
betaalbaar: " +
        df.format(objUitstappie.berekenBedrag(kostePerPersoon,
kosteToerGids)));
    }
    else{
        JOptionPane.showMessageDialog(null, "Die aansoek van " +
        objUitstappie.getSkoolNaam() + " is onsuksesvol.");
    }
}
}
```

**BYLAAG F: OPLOSSING VIR VRAAG 3: JAVA**

```
//=====
// Kode wat voorsien is
//=====

int terminaal = 0;
int rigting = 0;
char arrKodes[] = {'W',
    'A',
    'S',
    'R',
    'X',
    'D',
    'H',
    'P',
    'T',
    'L'};
String[] arrAktiwiteite = {
    "Waterpark",
    "Akwarium",
    "See",
    "Restaurante",
    "Inkopies",
    "Duik",
    "Hulptonbank",
    "Pikkewynpark",
    "Haaitenk",
    "Dolfynvertonings"
};
String[][] arrAktKodes = {{ "DXWAT", "HRDST", "STWLP", "RDT" },
    { "SWA", "SRXD", "LWXH", "SHA" },
    { "WLSR", "AT", "DATX", "HW" } };

private void btnTerminaal1ActionPerformed(java.awt.event.ActionEvent evt) {
    terminaal = 0;
}

private void btnTerminaal2ActionPerformed(java.awt.event.ActionEvent evt) {
    terminaal = 1;
}

private void btnTerminaal3ActionPerformed(java.awt.event.ActionEvent evt) {
    terminaal = 2;
}

private void btnNoordActionPerformed(java.awt.event.ActionEvent evt) {
    rigting = 0;
}

private void btnSuidActionPerformed(java.awt.event.ActionEvent evt) {
    rigting = 1;
}

private void btnOosActionPerformed(java.awt.event.ActionEvent evt) {
    rigting = 2;
}

private void btnWesActionPerformed(java.awt.event.ActionEvent evt) {
    rigting = 3;
}

```

```
//=====
// Globale veranderlikes
//=====
    String[] arrRigtings = {
        "Noord",
        "Suid",
        "Oos",
        "Wes"
    };

=====
// Vraag 3.1
=====
private void btnVraag31ActionPerformed(java.awt.event.ActionEvent evt) {
    txaVraag3.append(String.format("%-15s%-10s%-10s%-10s%-10s\n", "",
        "Noord", "Suid", "Oos", "Wes"));
    for (int iRy = 0; iRy < 3; iRy++) {
        txaVraag3.append(String.format("%-15s", "Terminaal " + (iRy + 1)));
        for (int iKolom = 0; iKolom < 4; iKolom++) {
            txaVraag3.append(String.format("%-10s",
                arrAktKodes[iRy][iKolom]));
        }
        txaVraag3.append("\n");
    }
}

=====
// Vraag 3.2
=====
private void btnVraag32ActionPerformed(java.awt.event.ActionEvent evt) {
    txaVraag3.setText("Terminaal " + (terminaal + 1) + ", " +
arrRigtings[rigting] + "\n");
    String kodes = arrAktKodes[terminaal][rigting];
    for (int i = 0; i < kodes.length(); i++) {
        for (int j = 0; j < arrKodes.length; j++) {
            if (kodes.charAt(i) == arrKodes[j]) {
                txaVraag3.append(arrAktiwiteite[j] + "\n");
            }
        }
    }
}

=====
// Vraag 3.3
=====
private void btnVraag33ActionPerformed(java.awt.event.ActionEvent evt) {
    int indeks = cmbVraag3.getSelectedIndex();
    int tel = 0;
    txaVraag3.setText("Toegangsroetes na " + cmbVraag3.getSelectedItem() +
        "\n");
    for (int iRy = 0; iRy < 3; iRy++) {
        for (int iKolom = 0; iKolom < 4; iKolom++) {
            if (arrAktKodes[iRy][iKolom].indexOf(arrKodes[indeks]) >= 0) {
                txaVraag3.append("Terminaal " + (iRy + 1) + ", " +
                    arrRigtings[iKolom] + "\n");
                tel++;
            }
        }
    }
    txaVraag3.append("\nGetal toegangsroetes: " + tel);
}

```

```
=====
// Vraag 3.4
=====
```

```
private void btnVraag34ActionPerformed(java.awt.event.ActionEvent evt) {
    txaVraag3.setText("Opgedateerde inligting:\n\n");
    int indeks = cmbVraag3.getSelectedIndex();
    for (int iRy = 0; iRy < 3; iRy++) {
        for (int iKolom = 0; iKolom < 4; iKolom++) {
            if (arrAktKodes[iRy][iKolom].indexOf(arrKodes[indeks]) >= 0) {
                arrAktKodes[iRy][iKolom] =
                    arrAktKodes[iRy][iKolom].replaceAll(arrKodes[indeks] + "", "");
            }
        }
    }
    JOptionPane.showMessageDialog(null, "Die toegangsroetes na " +
        arrAktiwiteite[indeks] + " is
        gesluit.");
    btnVraag31.doClick();
}
```

**BYLAAG G: OPLOSSING VIR VRAAG 1: DELPHI**

```
=====
// Kode wat voorsien is
=====

unit Vraag1_U;

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
  Dialogs, StdCtrls, ExtCtrls, ComCtrls, Spin, DateUtils;

type
  TfrmVraag1 = class(TForm)
    Panel1: TPanel;
    Panel2: TPanel;
    GroupBox1: TGroupBox;
    Label1: TLabel;
    Label2: TLabel;
    Label3: TLabel;
    edtLengte: TEdit;
    edtBreedte: TEdit;
    edtHoogte: TEdit;
    btnVraag11: TButton;
    edtVolume: TEdit;
    GroupBox2: TGroupBox;
    btnVraag12: TButton;
    edtKoste: TEdit;
    GroupBox3: TGroupBox;
    btnVraag13: TButton;
    Label4: TLabel;
    edtVraag13: TEdit;
    GroupBox4: TGroupBox;
    btnVraag14: TButton;
    redVraag14: TRichEdit;
    GroupBox5: TGroupBox;
    btnVraag151: TButton;
    redVraag15: TRichEdit;
    rgpPryse: TRadioGroup;
    Label5: TLabel;
    Label6: TLabel;
    Label7: TLabel;
    edtLeeftyd: TEdit;
    lblVolume: TLabel;
    Label8: TLabel;
    Label9: TLabel;
    edtAanvangkoste: TEdit;
    edtInkomste: TEdit;
    btnVraag152: TButton;
    procedure btnVraag11Click(Sender: TObject);
    procedure btnVraag12Click(Sender: TObject);
    procedure btnVraag13Click(Sender: TObject);
    procedure btnVraag14Click(Sender: TObject);
    procedure btnVraag151Click(Sender: TObject);
    procedure FormActivate(Sender: TObject);
    procedure btnVraag152Click(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;
```

```
var
    frmVraag1: TfrmVraag1;

//=====
//Globale veranderlike (Vr1.1 & Vr1.2)
//=====

    rVolume: real;

implementation

{$R *.dfm}
=====
// Vraag 1.1
=====

procedure TfrmVraag1.btnVraag11Click(Sender: TObject);
var
    rLengte, rBreedte, rHoogte: real;
begin
    rLengte := StrToFloat(edtLengte.Text);
    rBreedte := StrToFloat(edtBreedte.Text);
    rHoogte := StrToFloat(edtHoogte.Text);
    rVolume := rLengte * rBreedte * rHoogte;
    edtVolume.Text := FloatToStrF(rVolume, ffFixed, 6,2);
end;
=====
// Vraag 1.2
=====

procedure TfrmVraag1.btnVraag12Click(Sender: TObject);
var
    rKoste: real;
begin
    if rVolume <= 500 then
        rKoste := rVolume * 0.25
    else if rVolume <= 800 then
        rKoste := 500 * 0.25 + (rVolume - 500) * 0.35
    else
        rKoste := 500 * 0.25 + 300 * 0.35 + (rVolume - 800) * 0.45;

    edtKoste.Text := FloatToStrF(rKoste, ffCurrency, 6, 2);
end;
=====
// Vraag 1.3
=====

procedure TfrmVraag1.btnVraag13Click(Sender: TObject);
var
    iLeeftydInMaande, iJare, iMaande: integer;
begin

    iLeeftydInMaande := StrToInt(edtLeeftyd.Text);
    iMaande := iLeeftydInMaande mod 12;
    iJare := (iLeeftydInMaande - iMaande) div 12;
    edtVraag13.Text := IntToStr(iJare) + ' jaar en ' + IntToStr(iMaande)
        + ' maande';
end;
```

```
=====
// Vraag 1.4
=====
```

```
procedure TfrmVraag1.btnVraag14Click(Sender: TObject);
var
  rOprigKoste, rJaarInkomste: real;
  iJaarNommer: integer;
begin
  redVraag14.Paragraph.TabCount := 2;
  redVraag14.Paragraph.Tab[0] := 40;
  redVraag14.Paragraph.Tab[1] := 120;

  redVraag14.Lines.Add('Jaar' + #9 + 'Inkomste' + #9 + 'Balans');

  rOprigKoste := StrToFloat(edtAanvangKoste.Text);
  iJaarNommer := 1;
  rJaarInkomste := StrToFloat(edtInkomste.Text);

  while rOprigKoste > 0 do
  begin
    rOprigKoste := rOprigKoste - rJaarInkomste;
    if rOprigKoste > 0 then
      redVraag14.Lines.Add(IntToStr(iJaarNommer) + #9 + FloatToStrF
        (rJaarInkomste, ffCurrency, 8, 2) + #9 + FloatToStrF
        (rOprigKoste, ffCurrency, 8, 2))
    else
      redVraag14.Lines.Add(IntToStr(iJaarNommer) + #9 + FloatToStrF
        (rJaarInkomste, ffCurrency, 8, 2) + #9 + 'Afbetaal');
      Inc(iJaarNommer);
      rJaarInkomste := rJaarInkomste * 1.1;
    end;
  end;

end;
```

```
=====
//Vraag 1.5.1
=====
```

```
procedure TfrmVraag1.btnVraag151Click(Sender: TObject);
var
  iDobbel1, iDobbel2: integer;
begin
  redVraag15.Lines.Clear;

  iDobbel1 := random(6) + 1;
  iDobbel2 := random(6) + 1;
  redVraag15.Lines.Add('Dobbelsteen 1 = ' + IntToStr(iDobbel1));
  redVraag15.Lines.Add('Dobbelsteen 2 = ' + IntToStr(iDobbel2));

  if ((iDobbel1 = iDobbel2 + 1) OR (iDobbel1 = iDobbel2 - 1)) then
  begin
    btnVraag152.Enabled := true;
    rgpPryse.Enabled := true;
  end
  else
  begin
    btnVraag152.Enabled := false;
    rgpPryse.Enabled := false;
  end;
end;

end;
```

```
=====
//Vraag 1.5.2
=====
```

```
procedure TfrmVraag1.btnVraag152Click(Sender: TObject);
var
  sKaartjieNom, sVerwysNom: String;
  dDatum: TDateTime;
begin
  sKaartjieNom := inputbox('Toevoer van kaartjienommer', 'Sleutel jou kaartjie
se nommer is', '');
  dDatum := Date;
  sVerwysNom := sKaartjieNom + '#' + DateToStr(dDatum);
  sVerwysNom := sVerwysNom + '#' + Uppercase
  (copy(rgpPryse.Items[rgpPryse.ItemIndex], 1, 2));
  redVraag15.Lines.Add('Verwysingsnommer: ' + #13 + sVerwysNom);
end;
```

```
=====
//Kode wat voorsien is
=====
```

```
procedure TfrmVraag1.FormActivate(Sender: TObject);
begin
  CurrencyString := 'R';
  btnVraag152.Enabled := false;
  rgpPryse.Enabled := false;

end;

end.
```

**BYLAAG H: OPLOSSING VIR VRAAG 2: DELPHI****OBJEKKLAS:**

```

unit Uitstappie_U;

interface

uses SysUtils, Math, Messages, Dialogs, DateUtils;

Type
  TUITstappie = class(TObject)
  private
    { private declarations }
    fSkoolNaam, fBesoekeDatum: string;
    fGroepGrootte: integer;
    fToerGids: boolean;

  public
    { public declarations }
    constructor Create(sSkoolNaam:string; sDatum:string; iGroepGrootte:integer;
bToerGids:boolean);
    procedure setBesoekeDatum(sDatumVanBesoeke:string);
    function vereisToerGids: string;
    function isBevestig(iDagTotaal:integer): boolean;
    function berekenBedrag(rPersoonKoste,rGidsKoste:real):real;
    function toString: string;
    function getSkoolNaam:string;
    function getGroepGrootte:integer;
    function getBesoekeDatum: string;

  end;

implementation

{ TUITstappie }

//=====
// Kode wat voorsien is vir constructor
//=====

constructor TUITstappie.Create(sSkoolNaam, sDatum: string; iGroepGrootte:
integer;
  bToerGids: boolean);
begin
  fSkoolNaam := sSkoolNaam;
  fBesoekeDatum:= sDatum;
  fGroepGrootte:= iGroepGrootte;
  fToerGids := bToerGids;
end;
=====
// Vraag 2.1.1
=====

procedure TUITstappie.setBesoekeDatum(sDatumVanBesoeke: string);
begin
  fBesoekeDatum := sDatumVanBesoeke;
end;

```

```
=====
// Vraag 2.1.2
=====
function TUITstappie.vereisToergids: string;
begin
    if fToerGids then
        Result := 'Ja'
    else
        Result := 'Nee';
end;
=====
// Vraag 2.1.3
=====
function TUITstappie.isBevestig(iDagTotaal: integer): boolean;
begin
    if ((iDagTotaal + fGroepGrootte)<=500) then
        Result := true
    else
        Result := false;
end;
=====
// Vraag 2.1.4
=====
function TUITstappie.berekenBedrag(rPersoonKoste,rGidsKoste:real): real;
var
    iAantalGratis : integer;
    rBedrag : real;
begin
    iAantalGratis := fGroepGrootte div 10;
    rBedrag := (fGroepGrootte - iAantalGratis) * rPersoonKoste;
    if fToerGids then
        rBedrag := rBedrag + rGidsKoste;
    Result := rBedrag;
end;
=====
// Vraag 2.1.5
=====
function TUITstappie.toString: string;
begin
    Result := 'Skool se naam: ' + fSkoolNaam + #13 + 'Datum van besoek: ' +
        fBesoekDatum+#13+'Getal leerders: ' + IntToStr(fGroepGrootte) + #13
        +'Toergids: ' + vereisToergids;
end;
=====
// Kode wat voorsien is
=====
function TUITstappie.getSkoolNaam: string;
begin
    Result := fSkoolNaam;
end;

function TUITstappie.getGroepGrootte: integer;
begin
    Result := fGroepGrootte;
end;

function TUITstappie.getBesoekDatum: string;
begin
    Result := fBesoekDatum;
end;

end.
```

**HOOFVORMEENHEID: VRAAG2\_U.PAS**

```
unit Vraag2_U;

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
  Dialogs, ExtCtrls, StdCtrls, Uitstappie_U, ComCtrls;

type
  TfrmVraag2 = class(TForm)
    Panel1: TPanel;
    GroupBox1: TGroupBox;
    edtSkoolnaam: TEdit;
    edtGroepGrootte: TEdit;
    chbToergids: TCheckBox;
    btnVraag221: TButton;
    Label1: TLabel;
    Label2: TLabel;
    Label3: TLabel;
    lstDatums: TListBox;
    GroupBox3: TGroupBox;
    btnVraag222: TButton;
    pnlBeskikbaarheid: TPanel;
    cmbBeskikbareDatums: TComboBox;
    btnVraag223: TButton;
    Label4: TLabel;
    Panel2: TPanel;
    procedure btnVraag221Click(Sender: TObject);
    Function bepaalDagTotaal(sDatumVanBesoek: string): integer;
    procedure btnVraag222Click(Sender: TObject);
    procedure FormActivate(Sender: TObject);
    procedure btnVraag223Click(Sender: TObject);

  private
    { Private declarations }
  public
    { Public declarations }
  end;
const
  rKostePerPersoon= 75.00;
  rToerGids = 300.00;
var
  frmVraag2: TfrmVraag2;
  objUitstappie: TUITstappie;

implementation

{$R *.dfm}
=====
// Vraag 2.2.1
=====
procedure TfrmVraag2.btnVraag221Click(Sender: TObject);
var
  sSkoolNaam, sDatum: string;
  iGroepGrootte: integer;
  bToerGids: boolean;
begin
  sSkoolNaam := edtSkoolNaam.Text;
  sDatum := lstDatums.Items[lstDatums.ItemIndex];
  iGroepGrootte := StrToInt(edtGroepGrootte.Text);
```

```

if chbToerGids.Checked then
  bToerGids := true
else
  bToerGids := false;

  objUitstappie := TUITstappie.Create(sSkoolNaam, sDatum, iGroepGrootte,
bToerGids);
  ShowMessage('Uitstappie-objek is instansieer.');
```

```

=====
// Vraag 2.2.2 (a)
=====
```

```

function TfrmVraag2.bepaalDagTotaal(sDatumVanBesoek: string): integer;
var
```

```

  sSkoolNaam, sDatum: string;
  iGroepGrootte, iTotaal: integer;
  bToerGids: boolean;
  teksLeer: TextFile;
  sReel: string;
```

```
begin
```

```

  if not FileExists('DataV2.txt') then
  begin
    MessageDlg('Leer bestaan nie.', mtError, [mbOk], 0);
    Exit;
```

```

  end;
  iTotaal := 0;
  AssignFile(teksLeer, 'DataV2.txt');
  Reset(teksLeer);
```

```

  while NOT EOF(teksLeer) do
  begin
    readln(teksLeer, sReel);
    sSkoolNaam := copy(sReel, 1, pos('#', sReel) - 1);
    Delete(sReel, 1, pos('#', sReel));
    sDatum := copy(sReel, 1, pos('#', sReel) - 1);
    Delete(sReel, 1, pos('#', sReel));
    iGroepGrootte := StrToInt(sReel);
    if sDatum = sDatumVanBesoek then
      iTotaal := iTotaal + iGroepGrootte;
  end; // while
  CloseFile(teksLeer);
  Result := iTotaal;
```

```
end;
```

```

=====
// Vraag 2.2.2 (b)
=====
```

```

procedure TfrmVraag2.btnVraag222Click(Sender: TObject);
```

```
var
```

```

  I, iTotaal: integer;
  sDatum: string;
  bSuksesvol: boolean;
```

```
begin
```

```

  bSuksesvol := false;
  iTotaal := bepaalDagTotaal(objUitstappie.getBesoekDatum);
  if objUitstappie.isBevestig(iTotaal) then
  begin
    ShowMessage(objUitstappie.toString + #13 + 'Totale bedrag: ' + FloatToStrF
      (objUitstappie.berekenBedrag(rKostePerPersoon,rToerGids), ffCurrency,
8, 2));
```

```

    bSuksesvol := true;
  end
```

```
else
begin
  ShowMessage('Daar is nie plek op die datum wat gekies is nie.');
```

---

```
  cmbBeskikbareDatums.Clear;
  pnlBeskikbaarheid.Show;
  for I := 0 to 4 do
  begin
    sDatum := lstDatums.Items[I];
    iTotaal := bepaalDagTotaal(sDatum);
    if objUitstappie.isBevestig(iTotaal) then
    begin
      cmbBeskikbareDatums.Items.Add(sDatum);
    end;
  end;
end;
end;
=====
// Vraag 2.2.3
=====
procedure TfrmVraag2.btnVraag223Click(Sender: TObject);
var
  sDatum: string;

begin
  if cmbBeskikbareDatums.Items.Count > 0 then
  begin
    objUitstappie.setBesoekDatum
      (cmbBeskikbareDatums.Items[cmbBeskikbareDatums.ItemIndex]);
    ShowMessage(objUitstappie.toString + #13 + 'Bedrag betaalbaar: ' +
      FloatToStrF
        (objUitstappie.berekenBedrag(rKostePerPersoon,rToerGids), ffCurrency,
      8, 2));
  end
  else
  begin
    ShowMessage('Die aansoek van ' + objUitstappie.getSkoolnaam +
      ' was onsuksesvol.');
```

---

```
  end;
end;
=====
// Kode wat voorsien is
=====
procedure TfrmVraag2.FormActivate(Sender: TObject);
begin
  pnlBeskikbaarheid.Hide;
  CurrencyString := 'R';
end;

end.
```

**BYLAAG I: OPLOSSING VIR VRAAG 3: DELPHI**

```
unit Vraag3_U;

interface
uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
  Dialogs, StdCtrls, Buttons, ExtCtrls, ComCtrls;

type
  TfrmVraag3 = class(TForm)
    Panell: TPanel;
    GroupBox1: TGroupBox;
    btnTerminaal1: TBitBtn;
    btnTerminaal2: TBitBtn;
    btnTerminaal3: TBitBtn;
    GroupBox2: TGroupBox;
    btnNoord: TBitBtn;
    btnSuid: TBitBtn;
    btnOos: TBitBtn;
    btnWes: TBitBtn;
    GroupBox3: TGroupBox;
    btnVraag31: TButton;
    btnVraag32: TButton;
    btnVraag33: TButton;
    btnVraag34: TButton;
    redVraag3: TRichEdit;
    cmbVraag33: TComboBox;
    GroupBox4: TGroupBox;
    procedure btnVraag31Click(Sender: TObject);
    procedure FormActivate(Sender: TObject);
    procedure btnVraag32Click(Sender: TObject);
    procedure btnTerminaal1Click(Sender: TObject);
    procedure btnTerminaal2Click(Sender: TObject);
    procedure btnTerminaal3Click(Sender: TObject);
    procedure btnNoordClick(Sender: TObject);
    procedure btnSuidClick(Sender: TObject);
    procedure btnOosClick(Sender: TObject);
    procedure btnWesClick(Sender: TObject);
    procedure btnVraag33Click(Sender: TObject);
    procedure btnVraag34Click(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;
=====
// Kode wat voorsien is
=====
var
  frmVraag3: TfrmVraag3;
  arrKodes: array [1 .. 10] of char = (
    'W',
    'A',
    'S',
    'R',
    'X',
    'D',
    'H',
    'P',
    'T',
    'L'
  );
```

```

arrAktiwiteite: array [1 .. 10] of String = (
  'Waterpark',
  'Akwarium',
  'See',
  'Restaurante',
  'Inkopies',
  'Duik',
  'Hulptoonbank',
  'Pikkewynpark',
  'Haaitenk',
  'Dolfynvertonings'
);
arrAktKodes: array [1 .. 3, 1 .. 4] of String = (('DXWAT', 'HRDST',
  'STWLP', 'RDT'), ('SWA', 'SRXD', 'LWXH', 'SHA'),
  ('WLSR', 'AT', 'DATX', 'HW'));

iTerminaal: integer = 1;
iRigting: integer = 1;
//=====
// Globale veranderlikes
//=====
arrRigtings: array [1 .. 4] of string = (
  'Noord',
  'Suid',
  'Oos',
  'Wes'
);
=====
// Kode wat voorsien is
=====
implementation

{$R *.dfm}
procedure TfrmVraag3.btnTerminaal1Click(Sender: TObject);
begin
  iTerminaal := 1;
end;

procedure TfrmVraag3.btnTerminaal2Click(Sender: TObject);
begin
  iTerminaal := 2;
end;

procedure TfrmVraag3.btnTerminaal3Click(Sender: TObject);
begin
  iTerminaal := 3;
end;

procedure TfrmVraag3.btnNoordClick(Sender: TObject);
begin
  iRigting := 1;
end;

procedure TfrmVraag3.btnSuidClick(Sender: TObject);
begin
  iRigting := 2;
end;

procedure TfrmVraag3.btnOosClick(Sender: TObject);
begin
  iRigting := 3;
end;

```

```
procedure TfrmVraag3.btnWesClick(Sender: TObject);
begin
  iRigting := 4;
end;
=====
//Vraag 3.1
=====
procedure TfrmVraag3.btnVraag31Click(Sender: TObject);
var
  iRy, iKolom: integer;
  sReel: String;
begin
  redVraag3.Lines.Add(' ' + #9 + 'Noord' + #9 + 'Suid' + #9 +
'Oos' + #9 + 'Wes');
  for iRy := 1 to 3 do
  begin
    sReel := 'Terminaal ' + IntToStr(iRy) + #9;
    for iKolom := 1 to 4 do
    begin
      sReel := sReel + arrAktKodes[iRy, iKolom] + #9;
    end;
    redVraag3.Lines.Add(sReel);
  end;
end;

procedure TfrmVraag3.FormActivate(Sender: TObject);
begin
  redVraag3.Paragraph.TabCount := 4;
  redVraag3.Paragraph.Tab[0] := 80;
  redVraag3.Paragraph.Tab[1] := 130;
  redVraag3.Paragraph.Tab[2] := 180;
  redVraag3.Paragraph.Tab[3] := 230;
end;
=====
//Vraag 3.2
=====
procedure TfrmVraag3.btnVraag32Click(Sender: TObject);
Var
  i, j: integer;
  sKodes: string;
begin
  redVraag3.Clear;
  redVraag3.Lines.Add('Terminaal ' + IntToStr(iTerminaal) + ', ' + arrRigtings
[iRigting]);
  sKodes := arrAktKodes[iTerminaal, iRigting];
  for i := 1 to length(sKodes) do
  begin
    for j := 1 to length(arrKodes) do
    begin
      if sKodes[i] = arrKodes[j] then
        redVraag3.Lines.Add(arrAktiwiteite[j]);
    end;
  end;
end;
end;
```

```
=====
//Vraag 3.3
=====
```

```
procedure TfrmVraag3.btnVraag33Click(Sender: TObject);
var
  iRy, iKolom, iTel, iIndeks: integer;
begin
  redVraag3.Clear;
  iTel := 0;
  iIndeks := cmbVraag33.ItemIndex;
  redVraag3.Lines.Add('Toegangsroetes na ' +
cmbVraag33.Items[cmbVraag33.ItemIndex]
  );
  for iRy := 1 to 3 do
    for iKolom := 1 to 4 do
      if pos(arrKodes[iIndeks + 1], arrAktKodes[iRy, iKolom]) > 0 then
        begin
          redVraag3.Lines.Add('Terminaal ' + IntToStr(iRy) + ', ' + arrRigtings
            [iKolom]);
          Inc(iTel);
        end;
      redVraag3.Lines.Add(#13 + 'Getal toegangsroetes: ' + IntToStr(iTel));
    end;
end;
```

```
=====
//Vraag 3.4
=====
```

```
procedure TfrmVraag3.btnVraag34Click(Sender: TObject);
var
  iIndeks, iRy, iKolom: integer;
begin
  redVraag3.Clear;
  redVraag3.Lines.Add('Opedateerde inligting:');
  redVraag3.Lines.Add('');
  iIndeks := cmbVraag33.ItemIndex;
  for iRy := 1 to 3 do
    for iKolom := 1 to 4 do
      if pos(arrKodes[iIndeks + 1], arrAktKodes[iRy, iKolom]) > 0 then
        begin
          Delete(arrAktKodes[iRy, iKolom], pos(arrKodes[iIndeks + 1],
            arrAktKodes[iRy, iKolom]), 1);
        end;

        ShowMessage('Die toegangsroetes na ' + arrAktiwiteite[iIndeks + 1]
          + ' is gesluit.');
```

```
    btnVraag31.Click;
  end;

end.
```